



Microsoft

70-357 Exam

Microsoft Developing Mobile Apps Exam

Thank you for Downloading 70-357 exam PDF Demo

You can Buy Latest 70-357 Full Version Download

<https://www.certkillers.net/Exam/70-357>

<https://www.certkillers.net>

Version: 9.0

Case Study: 1

Background

Business requirements

In this section, you will see one or more sets of questions with the same scenario and problem. Each question presents a unique solution to the problem, and you must determine whether the solution meets the slated goals. Any of the solutions might solve the problem. It is also possible that none o' the solutions solve the problem.

Technical requirements

Application structure

Once you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appe.ir in the review screen.

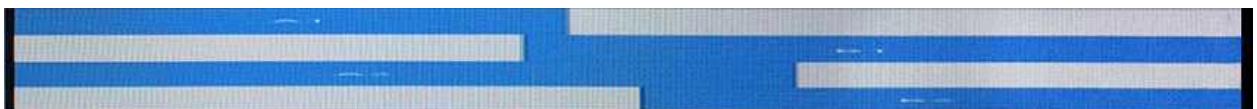
The timeline element of the app has the following layout requirements:

- the timeline must adapt to the screen size and orientation of the device.
- The timeline size must dynamically change if the window containing the content is resized by the user.
- The user must be able to scroll through the timeline horizontally when the device is in landscape mode.
- The user must be able to scroll through the timeline vertically when the device is in portrait mode.
- The timeline must begin scrolling as soon as a scroll is detected. Scrolling must continue for a short distance after the scroll input has stopped.
- Scroll bars or panning controls must always be visible.

The following image depicts the layout for the timeline section of the app when the device is using landscape orientation:



the following image depicts the layout tor the timeline section of the app when the device is using portrait orientation:



The content element of the app has the following layout requirements:

When a user selects an item on the timeline, the details for that item must display beneath or to the right of the timeline

- The content section must display one page of information. The element must be a child of the selected item in the timeline.
- Users must be able to return to a previously selected event by pressing the Back button.

the user must be able to navigate the application using the interface below:



- The Favorite button mark the Current content to be displayed in a Favorites panel.
- The Back and Forward buttons navigate through the app selection history. Both buttons must be available on all devices.
- The Note button allows the user to manage notes about the current content.
- The app must support touch, mouse, and stylus input.
- The app layout must automatically adapt to the screen size and orientation.

Layout Requirement:

You identify the following layout requirements:

General

- All user interface (UI) elements must continuously scale when a user resizes the window.
- UI controls must be smaller and spaced closer together if there is a mouse or stylus available
- UI controls must be larger and spaced farther apart if the device supports touch and there is no mouse or pointer available

Timeline

- The timeline must be displayed in a horizontal layout when the device is in a landscape orientation or

When the horizontal width is greater than the vertical height.

- The timeline must be displayed in a vertical layout when the device is in a portrait orientation or when the vertical height is greater than the horizontal width.
- Each item in the past must be linked to the next item in the future
- User must be able to scroll from past events to future events or from future events to past events.
- The app must only allow one level of detail to be linked to each item in the timeline

New Tab:

You must optimize the app using the following guidelines:

- You must minimize the time it takes to display content when an item on the timeline is selected.
- The app must respect memory and resource constraints for all devices.

XML coding style:

All code and markup must conform to the following style guidelines:

- Use resource dictionaries for styles that are used more than once.
- Limit the use of nested panels.
- Use built-in properties of existing panels instead of using separate style objects.
- Use the navigation structure that best models the data without exceeding the requirements of the app.

MainPage.xaml

Relevant portions of the app files are shown below, (line numbers in the code segments are included for reference only and include a two character prefix that denotes the specific file to which they belong.)

```

MP01 <Page
      x:Class="_70357rm.MainPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:local="using:_70357rm"
      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
      xmlns:m="http://schemas.openxmlformats.org/markup-compatibility/2006"
      mc:Ignorable="d">
MP02     <Grid Background="{ThemeResource ApplicationPageBackgroundThemeBrush}">
MP03
MP04         <RelativePanel BorderBrush="Gray" BorderThickness="10">
MP05             <Rectangle x:Name="A1" Fill="Red" MinHeight="200" MinWidth="400"
                  RelativePanel.AlignLeftWithPanel="True"
                  RelativePanel.AlignTopWithPanel="False" />
MP06             <Rectangle x:Name="B1" Fill="Blue" MinHeight="200" MinWidth="400"
                  RelativePanel.Below="A1"
                  RelativePanel.RightOf="A1"
                  RelativePanel.AlignRightWithPanel="True"
                  RelativePanel.AlignBottomWithPanel="False" />
MP07         </RelativePanel>
MP08     </Grid>
MP09 </Page>

```

Relevant portions of the app files are shown below, (Line numbers in the code segments are included for reference only and include a two character prefix that denotes the specific file to which they belong.)

```

AS01 <Page
    x: Class = "_70357rm.Settings"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local="using:_70357rm"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:m="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d">
AS02     <Grid Background="AliceBlue">
AS03         <Border BorderBrush="DarkBlue" BorderThickness="5" />
AS04         <Grid Margin="5 5 5 5">
AS05             <StackPanel HorizontalAlignment="Center">
AS06                 <TextBlock Text="Date Settings" Foreground="DarkBlue" FontFamily="Arial"
FontSize="20" FontStyle="Normal"
                    FontWeight="Bold" Margin="0 5 0 5" HorizontalAlignment="Center" />
AS07                 <StackPanel Orientation="Horizontal">
AS08                     <CheckBox Content="Center on Date" FontFamily="Arial" FontSize="14"
FontStyle="Normal" Margin="20,0,0,0" />
AS09                     <CheckBox Content="Set Start Date" FontFamily="Arial" FontSize="14"
FontStyle="Normal" Margin="20,0,0,0" />
AS10                 </StackPanel>
AS11                 <TextBlock Text="Start Date" Foreground="DarkBlue" FontFamily="Arial"
FontSize="20" FontStyle="Normal"
                    FontWeight="Bold" Margin="0 5 0 5" HorizontalAlignment="Center" />
AS12                 <StackPanel Orientation="Horizontal">
AS13                     <TextBlock Text="Month:" Width="75" />
AS14                     <TextBox Width="200" />
AS15                 </StackPanel>
AS16                 <StackPanel Orientation="Horizontal">
AS17                     <TextBlock Text="Day:" Width="75" />
AS18                     <TextBox Width="200" />
AS19                 </StackPanel>
AS20                 <StackPanel Orientation="Horizontal">
AS21                     <TextBlock Text="Year:" Width="75" />
AS22                     <TextBox Width="200" />
AS23                 </StackPanel>
AS24                 <Ellipse Fill="Blue" Width="50" Height="50" Margin="0 5 0 5" />
AS25                 <TextBlock FontFamily="Arial" FontSize="14" Foreground="White" Text="Save"
Margin="127 -38 0 0" />
AS26             </StackPanel>
AS27         </Grid>
AS28     </Grid>
AS29 </Page>
AS30 </Page>

```

Relevant portions of the app files are shown below.(Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)


```

01 <ResourceDictionary
02     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
03     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
04     xmlns:local="using:_70357rm">
05     <Style x:Key="fill">
06         <Setter Property="Foreground" Value="DarkBlue"/>
07     </Style>
08     <Style x:Key="text">
09         <Setter Property="FontFamily" Value="Arial"/>
10     </Style>
11     <Style x:Key="big">
12         <Setter Property="FontSize" Value="20"/>
13     </Style>
14     <Style x:Key="small">
15         <Setter Property="FontSize" Value="14"/>
16     </Style>
17     <Style x:Key="strong">
18         <Setter Property="FontWeight" Value="Bold"/>
19     </Style>
20     <Style x:Key="light">
21         <Setter Property="FontWeight" Value="Normal"/>
22     </Style>
23     <Style x:Key="normal">
24         <Setter Property="FontStyle" Value="Normal"/>
25     </Style>
26     <Style x:Key="pad">
27         <Setter Property="Margin" Value="0 5 0 5"/>
28     </Style>
29     <Style x:Key="gap">
30         <Setter Property="Margin" Value="20 0 0 0"/>
31     </Style>
32     <Style x:Key="middle">
33         <Setter Property="HorizontalAlignment" Value="Center"/>
34     </Style>
35     <Style TargetType="CheckBox" x:Key="check">
36         <Setter Property="FontFamily" Value="Arial" />
37         <Setter Property="FontSize" Value="14" />
38         <Setter Property="FontStyle" Value="Normal" />
39         <Setter Property="Margin" Value="20 0 0 0" />
40     </Style>
41     <Style TargetType="TextBox" x:Key="heading">
42         <Setter Property="Foreground" Value="DarkBlue" />
43         <Setter Property="FontFamily" Value="Arial" />
44         <Setter Property="FontSize" Value="20" />
45         <Setter Property="FontStyle" Value="Normal" />
46         <Setter Property="FontWeight" Value="Bold" />
47         <Setter Property="Margin" Value="0 5 0 5" />
48         <Setter Property="HorizontalAlignment" Value="Center" />
49     </Style>
50 </ResourceDictionary>

```

New Tab:

Relevant portions of the app files are shown below.(Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
MX01 private void App_BackRequest(object sender, Windows.UI.Core.BackRequestedEventArgs e)
MX02     {
MX03         Frame page = Window.Current.Content as Frame;
MX04         if ( page != null)
MX05         {
MX06             if ( page.CanGoBack )
MX07             {
MX08                 page.GoBack();
MX09             }
MX10         }
MX11     }
```

Question: 1

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains n unique solution. Determine whether the solution meets the stated goals.

You need to implement the appropriate XAML layout for the Timeline app.

Solution: You create an instance of a StackPanel class.

Does this meet the goal?

- A. Yes
- B. No

Answer: B

Explanation:

StackPanel is a simple layout panel that arranges its child elements into a single line that can be oriented horizontally or vertically. StackPanel controls are typically used in scenarios where you want to arrange a small subsection of the UI on your page.

The following XAML shows how to create a vertical StackPanel of items.

XAML

```
<StackPanel>
<Rectangle Fill="Red" Height="44"/>
<Rectangle Fill="Blue" Height="44"/>
<Rectangle Fill="Green" Height="44"/>
<Rectangle Fill="Orange" Height="44"/>
</StackPanel>
```

The result looks like this.



Question: 2

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains n unique solution. Determine whether the solution meets the stated goals.

You need to implement the appropriate XAML layout for the Timeline app.

Solution: You create an instance of a SplitView control.

Does this meet the goal?

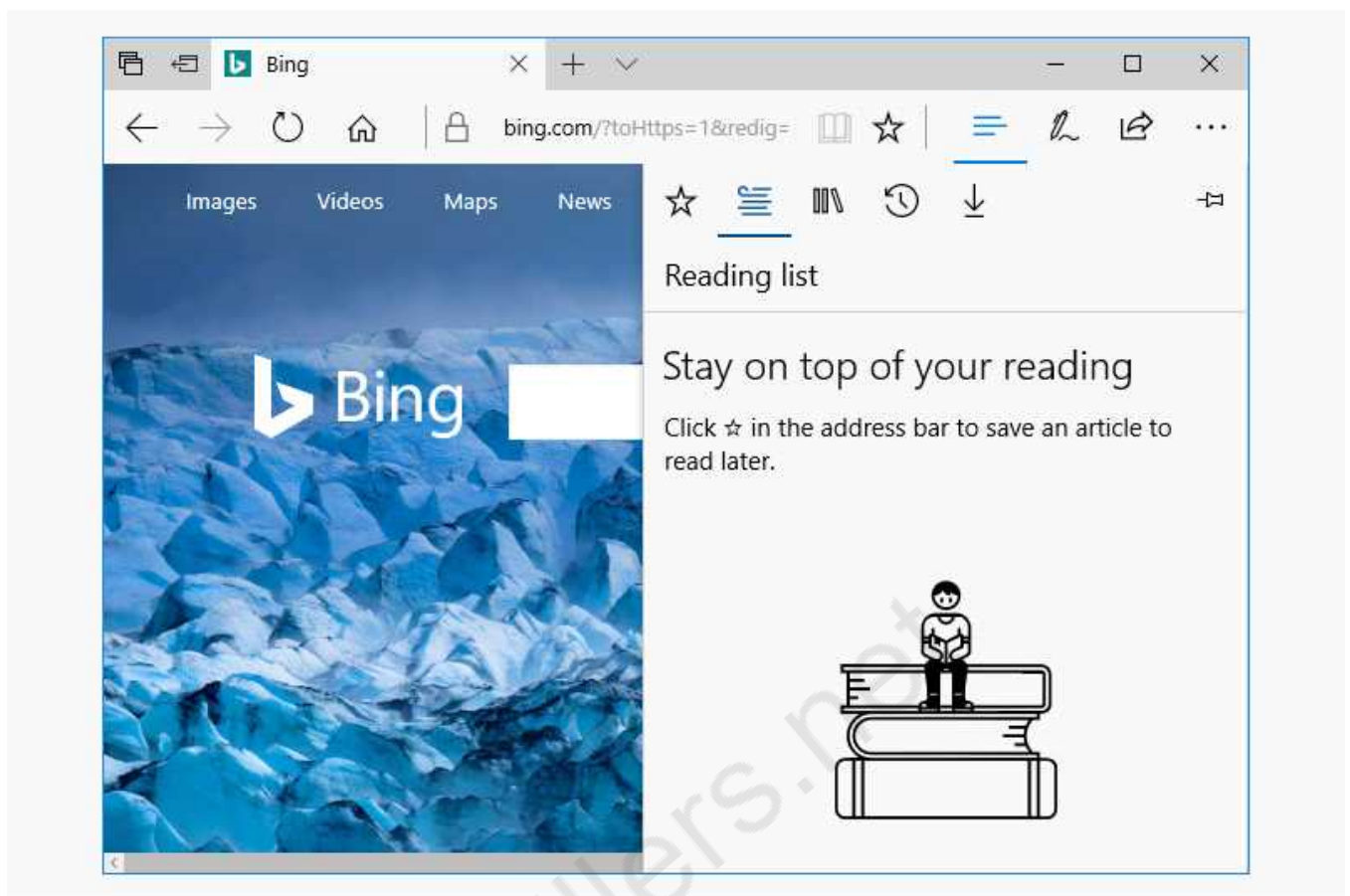
- A. Yes
- B. No

Answer: B

Explanation:

A split view control has an expandable/collapsible pane and a content area.

Here is an example of the Microsoft Edge app using SplitView to show its Hub.



Question: 3

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You need to implement the appropriate XAML layout (or the Timeline app).

Solution: You create an instance of a RelativePanel class.

Does this meet the goal?

- A. Yes
- B. No

Answer: A

RelativePanel lets you layout UI elements by specifying where they go in relation to other elements and in relation to the panel. By default, an element is positioned in the upper left corner of the panel.

Question: 4

DRAG DROP

You need to create the user interface for the timeline.

Which four markup segments should you use to develop the solution? To answer, move the appropriate markup segments from the list of markup segments to the answer area and arrange them in the correct order.

NOTE: More than one order of answer choices is correct. You will receive credit for any of the correct orders you select.

Markup segments

```
<Button Content ="Favorite"
Click ="UiButton_Click"/>
<TextBlock Text ="|" Margin =
"0 2 0 2"/>
<Button Content ="Back" Click =
"UiButton_Click"/>
<Button Content = "Forward"
Click ="UiButton_Click"/>
```

```
<AppBar>
```

```
</CommandBar>
```

```
<CommandBar>
```

```
<AppBarButton Icon ="Favorite"
Label ="Favorite"
Click="UiButton_Click"/>
<AppBarSeparator />
<AppBarButton Icon = "Back"
Label = "Back"
Click="UiButton_Click" />
<AppBarButton Icon ="Forward"
Label = "Forward"
Click="UiButton_Click"/>
```

```
<AppBar.Tag>
<Button Content = "Comment"
Click="UiButton_Click"/>
</AppBar.Tag>
</AppBar.Content>
<TextBlock Text ="Navigation"
Margin ="12,14"/>
</AppBar.Content>
```

```
<CommandBar.SecondaryCommands>
<AppBarButton Icon="Comment"
Label = "Notes"
Click="UiButton_Click"/>
</CommandBar.SecondaryCommands>
<CommandBar.Content>
<TextBlock Text = "Navigation"
Margin ="12,14"/>
</CommandBar.Content>
```

```
</AppBar>
```

Answer Area



Answer:

Answer Area

```

<CommandBar>

<AppBarButton Icon="Favorite"
Label="Favorite"
Click="UiButton_Click"/>
<AppBarSeparator />
<AppBarButton Icon="Back"
Label="Back"
Click="UiButton_Click" />
<AppBarButton Icon="Forward"
Label="Forward"
Click="UiButton_Click"/>

<CommandBar.SecondaryCommands>
  <AppBarButton Icon="Comment"
Label="Notes"
Click="UiButton_Click"/>
</CommandBar.SecondaryCommands>
<CommandBar.Content>
  <TextBlock Text="Navigation"
Margin="12,14"/>
</CommandBar.Content>

</CommandBar>

```

Box 1: <Commandbar>

Command bars (also called "app bars") provide users with easy access to your app's most common tasks, and can be used to show commands or options that are specific to the user's context, such as a photo selection or drawing mode. They can also be used for navigation among app pages or between app sections. Command bars can be used with any navigation pattern.

XAML provides both the AppBar control and the CommandBar control. You should use the AppBar only when you are upgrading a Universal Windows 8 app that uses the AppBar, and need to minimize changes. For new apps in Windows 10, we recommend using the CommandBar control instead.

Box 2: <AppBarButton .. etc.

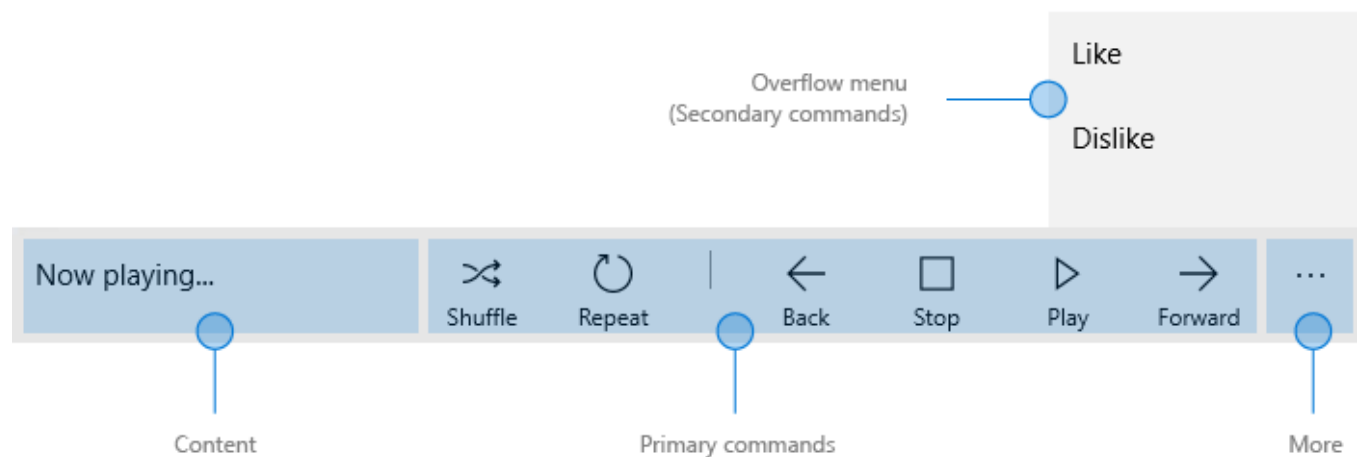
The CommandBar control is a general-purpose, flexible, light-weight control that can display both complex content, such as images or text blocks, as well as simple commands such as AppBarButton, AppBarToggleButton, and AppBarSeparator controls.

Box 3: <CommandBar.SecondaryCommands> etc.

The overflow menu is shown only when the command bar is open and the SecondaryCommands property is populated. The new dynamic overflow behavior will automatically move primary commands into the SecondaryCommands area when space is limited.

Box 4: </Commandbar>

Example: Here is a same command bar in its open state. The labels identify the main parts of the control.



This example creates the command bar shown above.

```
<CommandBar>
  <AppBarToggleButton Icon="Shuffle" Label="Shuffle" Click="AppBarButton_Click" />
  <AppBarToggleButton Icon="RepeatAll" Label="Repeat" Click="AppBarButton_Click"/>
  <AppBarSeparator/>
  <AppBarButton Icon="Back" Label="Back" Click="AppBarButton_Click"/>
  <AppBarButton Icon="Stop" Label="Stop" Click="AppBarButton_Click"/>
  <AppBarButton Icon="Play" Label="Play" Click="AppBarButton_Click"/>
  <AppBarButton Icon="Forward" Label="Forward" Click="AppBarButton_Click"/>
  <CommandBar.SecondaryCommands>
    <AppBarButton Icon="Like" Label="Like" Click="AppBarButton_Click"/>
    <AppBarButton Icon="Dislike" Label="Dislike" Click="AppBarButton_Click"/>
  </CommandBar.SecondaryCommands>
  <CommandBar.Content>
    <TextBlock Text="Now playing..." Margin="12,14"/>
  </CommandBar.Content>
</CommandBar>
```

Thank You for trying 70-357 PDF Demo

To Buy Latest 70-357 Full Version Download visit link below

<https://www.certkillers.net/Exam/70-357>

Start Your 70-357 Preparation

[Limited Time Offer] Use Coupon “CKNET” for Further discount on your purchase. Test your 70-357 preparation with actual exam questions.

<https://www.certkillers.net>